Statistical Techniques in Robotics (16-831, S21) Lecture #09 (Wednesday, March 3)

OGD & NormExpGD

Lecturer: Kris Kitani *Scribes: Andy Wei, Yi-Chun Chen*

1 Review

In the last lecture, we talked about online mirror descent (OMD). In this section, we will first give the review of OMD, dive into the definitions of Conjugate Function and Bregman Divergence, and finally lead to the analysis of OMD.

1.1 Online Mirror Descent

Online mirror descent is formulated from follow the regularized leader (FTRL)[2]. By reparametrizing as dual parameter and mirror function , when FTRL loss is linear, and regularization is convex, it is named as OMD. The idea is to optimize in the dual space and mirror in the primal space, as shown in Algorithm 1.

Algorithm 1 OMD (Convex set S, g: $\mathbb{R}^D \to S$)

for $t = 1, \cdots, T$ **do**
  RECEIVE($f^{(t)} : S \to \mathbb{R}$) ▷ $f^{(t)}(\cdot)$: time-varying loss function
  $\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \eta \boldsymbol{z}^{(t)}, \boldsymbol{z}^{(t)} \in \partial f^{(t)}(\boldsymbol{w}^{(t)})$ ▷ $\boldsymbol{\theta}$: parameters in dual space
  $\boldsymbol{w}^{(t+1)} = g(\boldsymbol{\theta}^{(t+1)})$ ▷ $g(\cdot)$: mirror function
end for

1.2 Conjugate Function

1.2.1 Primal Representation and Dual Representation

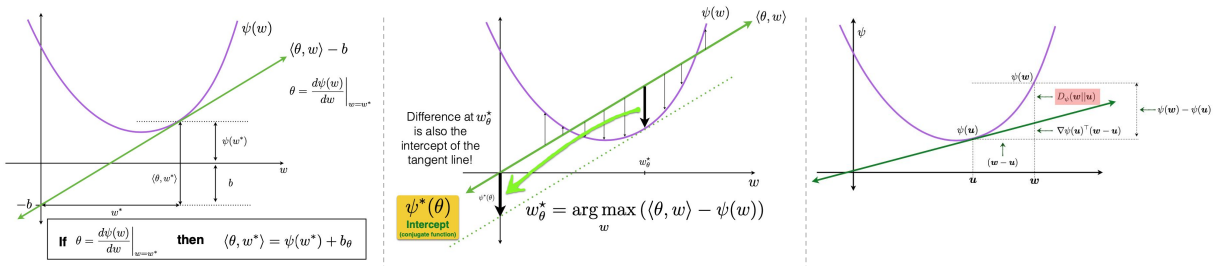
First we need to know parametrizations of a function in primal space and dual space:

$$Primal : \{\psi(w),\ w\} \qquad Dual : \{b(\theta),\ \theta\}$$

, where w and $\psi(w)$ are the parameter and function and θ and $b(\theta)$ are the slope and intercept. Then, let's look into the dual parametrization in dual space in the aspect of geometry.

In Figure 1 (a), we observe that when $\theta = \frac{d\psi(w)}{dw}\Big|_{w=w^*}$ holds, $\langle \theta,\ w^* \rangle = \psi(w^*) + b_\theta$. For a corresponding pair $\{\theta, w^*\}$, we rearrange the equation above and have

$$-b(\theta) = -\langle \theta, w^*(\theta) \rangle + \psi(w^*(\theta))$$



(a) Geometry of tangents in dual space (b) Geometry of conjugate (c) Geometry of bregman divergence

Figure 1: Geometry representation for review section.

1.2.2 Geometry of Conjugate Function

In Figure 1 (b), consider a function $\langle \theta, w \rangle - \psi(w)$, we can observe we can find a w_θ^* that maximizes the function. Also, at w_θ^*, the difference of the functions is also the intercept of the tangent line, $\psi^*(\theta)$. It is concluded that the conjugate function is, for any slope θ, $\psi^*(\theta) = \max_w(\langle \theta, w \rangle - \psi(w))$.

1.3 Bregman Divergence

The definition of Bregman Divergence is below; that is, the distance between two points according to some proximity function ψ, which in the context of OMD is the regularization function. Geometrically, as show in Figure 1, (c) it is the approximation error between a linear approximation of some convex function.

$$\boldsymbol{D}_\psi(\boldsymbol{w}||\boldsymbol{u}) = \psi(\boldsymbol{w}) - \psi(\boldsymbol{u}) - \nabla\psi(\boldsymbol{u})^T(\boldsymbol{w} - \boldsymbol{u})$$

1.4 OMD Analysis

We aim to analyze the regret bound of OMD. The loss of any arbitrary vector $\boldsymbol{u}$ is

$$\psi(\boldsymbol{u}) + \sum_{t=1}^{T} \boldsymbol{u} \cdot \boldsymbol{z}^{(t)} = \psi(\boldsymbol{u}) - \boldsymbol{u} \cdot \boldsymbol{\theta}^{(T+1)}$$

By applying Fenchel-Young Inequality $\psi^*(\boldsymbol{\theta}) \geq (\langle \boldsymbol{w}, \boldsymbol{\theta} \rangle - \psi(\boldsymbol{w}))$ to RHS, we get

$$\psi(\boldsymbol{u}) - \boldsymbol{u} \cdot \boldsymbol{\theta}^{(T+1)} \geq -\psi^*(\boldsymbol{\theta}^{(\boldsymbol{T+1})})$$

Next, expand the RHS with telescoping and we can derive the general OMD regret bound as follow:

$$R(\boldsymbol{u}) = \sum_{t=1}^{T} \boldsymbol{w}^{(t)} \cdot \boldsymbol{z}^{(t)} - \boldsymbol{u} \cdot \boldsymbol{z}^{(t)} \leq \psi(\boldsymbol{u}) - \psi(\boldsymbol{w}^{(1)}) + \sum_{t=1}^{T} D_{\psi^*}(-\boldsymbol{z}^{(1:t)}|| - \boldsymbol{z}^{(1:t-1)})$$

2 Summary

2.1 Online Gradient Descent (OGD)

2.1.1 Gradient Descent

Gradient descent [1] is the standard approach for minimizing differentiable convex functions. The gradient descent is listed as Algorithm 2 below. The line 4 specifies that the algorithm updates weight by gradient with a scaling factor η. The gradient, denoted a $\nabla f(\mathbf{w})$ in the line 3, is a gradient of a differentiable function $f : \mathbb{R}^N \to \mathbb{R}$ at $\mathbf{w}$. It is a vector of a parital derivatives

$$\nabla f(\mathrm{w}) = \{\frac{\partial f(\mathbf{w})}{\partial w_1}, \frac{\partial f(\mathbf{w})}{\partial w_2}, \cdots, \frac{\partial f(\mathbf{w})}{\partial w_N}\}$$

Algorithm 2 Gradient Descent (GD)

1: $\mathbf{w}^{(0)} \leftarrow 0$ ▷ Weight initialization
2: **for** $t = 1, \cdots, T$ **do**
3: COMPUTE $(\nabla f(\mathbf{w}^{(t-1)}))$ ▷ Compute gradient of current step w.r.t. weight
4: $\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} - \eta \nabla f(\mathbf{w}^{(t-1)})$ ▷ Update weight by gradient and learning rate η
5: **end for**

2.1.2 Understanding Gradient Descent

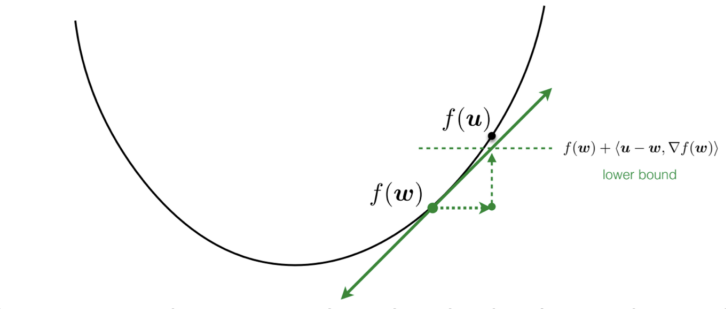
One may ask how the update equation comes from. There are 3 perspectives of viewing GD algorithm: geometric, linear approximation with regularization, and isometric quadratic approximation.
Geometric The first, most intuitive way is - moving in the direction opposite of the gradient. It is easy to say that when the learning rate is sufficiently small, we could minimize the function value by moving in such direction. However, such perspective is not very rigorous.
Linear Approximation with Regularization The second perspective comes from linear approximation. Given $\mathbf{w}$, we can approximate the $f(\mathbf{u})$ by first-order Taylor expansion.

$$f(\mathbf{u}) \approx f(\mathbf{w}) + \langle(\mathbf{u} - \mathbf{w}), \nabla f(\mathbf{w})\rangle$$

Our objective of minimizing function value f will then become

$$\min_u f(\mathbf{u}) \approx \min_u \{f(\mathbf{w}) + \langle(\mathbf{u} - \mathbf{w}), \nabla f(\mathbf{w})\rangle\}$$

However, directly minimize the above equation will lead to solution at negative infinity.



As $f(\mathbf{u}) \approx f(\mathbf{w}) + \langle (\mathbf{u} - \mathbf{w}), \nabla f(\mathbf{w}) \rangle$ only holds when $\mathbf{u}$ is close to $\mathbf{w}$, we can add an L2 regularization term.

$$\min_{\mathbf{u}} \|\mathbf{u} - \mathbf{w}\|_2^2$$

The full objective to minimize becomes

$$\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u} - \mathbf{w}\|_2^2 + \eta(f(\mathbf{w}) + \langle (\mathbf{u} - \mathbf{w}), \nabla f(\mathbf{w}) \rangle)$$

With switching the notation $\mathbf{u} \to \mathbf{w}$ and $\mathbf{w} \to \mathbf{w}^{(t)}$, the equation becomes

$$\mathbf{w}^{(t+1)} = \text{argmin}_{\mathbf{w}} \frac{1}{2}\|\mathbf{w} - \mathbf{w}^{(t)}\|_2^2 + \eta(f(\mathbf{w}^{(t)}) + \langle (\mathbf{w} - \mathbf{w}^{(t)}), \nabla f(\mathbf{w}^{(t)}) \rangle)$$

Be taking the first order derivative to find the optimal $\mathbf{w}$, we got the GD algorithm update rules.

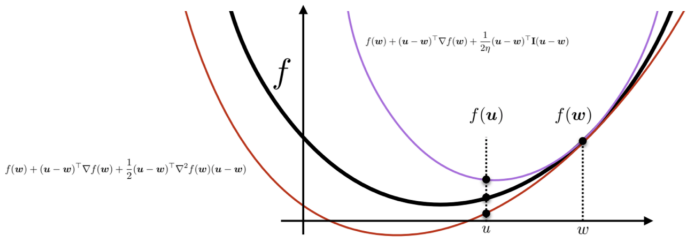
$$\frac{\partial \frac{1}{2}\|\mathbf{w} - \mathbf{w}^{(t)}\|_2^2 + \eta(f(\mathbf{w}^{(t)}) + \langle (\mathbf{w} - \mathbf{w}^{(t)}), \nabla f(\mathbf{w}^{(t)}) \rangle)}{\partial \mathbf{w}} = 0$$

$$(\mathbf{w} - \mathbf{w}^{(t)}) + \eta(0 + \nabla f(\mathbf{w}^{(t)}) = 0$$

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \nabla f(\mathbf{w}^{(t)})$$

Isometric Quadratic Approximation The third perspective is to approximate the function value using isometric quadratic approximation. Start with second order Taylor expansion

$$\min_u f(\mathbf{u}) \approx \min_u \{f(\mathbf{w}) + (\mathbf{u} - \mathbf{w})^T \nabla f(\mathbf{w}) + \frac{1}{2}(\mathbf{u} - \mathbf{w})^T \nabla^2 f(\mathbf{w})(\mathbf{u} - \mathbf{w})\}$$



Using isometric quadratic approximation,

$$\min_u f(\mathbf{u}) \approx \min_u \{f(\mathbf{w}) + (\mathbf{u} - \mathbf{w})^T \nabla f(\mathbf{w}) + \frac{1}{2\eta}(\mathbf{u} - \mathbf{w})^T \mathbf{I}(\mathbf{u} - \mathbf{w})\}$$

where η is a tunable variance parameter.
By multiplying the RHS objective function with η and rearranging terms, the objective function becomes

$$\min_u \frac{1}{2}(\|\mathbf{u}-\mathbf{w})\|_2^2 + \eta(f(\mathbf{w}) + (\mathbf{u}-\mathbf{w})^T \nabla f(\mathbf{w}))$$

By similar manner with the linear approximation perspective, replacing the notation $\mathbf{u} \to \mathbf{w}$ and $\mathbf{w} \to \mathbf{w}^{(t)}$, we have

$$\mathbf{w}^{(t+1)} = \text{argmin}_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}-\mathbf{w}^{(t)}\|_2^2 + \eta(f(\mathbf{w}^{(t)}) + \langle(\mathbf{w}-\mathbf{w}^{(t)}), \nabla f(\mathbf{w}^{(t)})\rangle)$$

The remaining step is taking the first order derivative to find minimum solution, which is exactly the same as the final step of linear quadratic approximation.

2.1.3 Stochastic Gradient Descent

From last subsection, we know that the update rule of gradient descent is

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \nabla f(\mathbf{w}^{(t)})$$

Most of the time calculating gradient $\nabla f(\mathbf{w})$ requires summing over all data points, which may not be feasible when dataset size is large. To speed up, one can replace the gradient by randomly sampled data points instead of exact gradient at each iteration; that leads to Stochastic Gradient Descent [3], listed as Algorithm 3.

Algorithm 3 Stochastic Gradient Descent (SGD)

1: $\mathbf{w}^{(0)} \leftarrow 0$ ▷ Weight initialization
2: $\eta > 0$
3: **for** $t = 1, \cdots, T$ **do**
4: $\quad z \sim \mathcal{D}$ ▷ Sampled mini-batch z from full dataset
5: $\quad \mathbf{v}^{(t)} = \nabla_z f(\mathbf{w}^{(t-1)})$ ▷ Compute gradient from sampled data instead of full data points
6: $\quad \mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} - \eta \mathbf{v}^{(t)})$ ▷ Update weight with $\mathbf{v}^{(t)}$ instead of exact gradient
7: **end for**

Different from GD, SGD sampled the mini-batch for calculating gradient at line 3. The gradient for update is calculated through sampled mini-batch at line 4 and is used for update at line 5. With such sampling, we can reduce the computation on gradient while remaining the expected value of gradient being the same as exact gradient at each iteration. One key observation of SGD is that the algorithm does not require the access of full dataset at each iteration, which looks like online convex optimization.

2.1.4 Online Gradient Descent(OGD) as Online Mirrow Descent (OMD)

We started from showing that the Online Gradient Descent algorithm is actually Online Mirrow Descent with linear loss and quadratic regularizer. Starting from defining the regularization function

$$\psi(\mathbf{w}) = \frac{1}{2\eta}\|\mathbf{w}\|_2^2$$

And the loss function

$$f(w) = \langle \mathbf{w}, \theta \rangle$$

Our update rule becomes

$$\mathbf{w}^{(t+1)} = \text{argmin}_{\mathbf{w}} \langle \mathbf{w}, -\theta^{(t+1)} \rangle + \frac{1}{2\eta}\|w\|_2^2$$

By solving the first order partial derivative equals to 0, we can find the minimizer

$$\mathbf{w}^{(t+1)} = -\eta\theta$$

which is the mirror function.
Algorithm 4 is the Online Sub-Gradient Descent, which adapted the OMD algorithm with the mirror function induced from linear loss function and quadratic regularizer.

Algorithm 4 Online Sub-Gradient Descent

1: **for** $t = 1, \cdots, T$ **do**
2: $\quad \theta^{(t+1)} = \theta^{(t)} - \eta\mathbf{z}$ ▷ $\mathbf{z}^{(t)} = \partial f^{(t)}(\mathbf{w}^{(t)})$
3: $\quad \mathbf{w}^{(t+1)} = -\eta\theta^{(t+1)}$ ▷ Mirror Projection
4: **end for**

From line 2 and 3 of Algorithm 4, we can further show that it is same ass GD update rule by

$$\begin{aligned}
\mathbf{w}^{(t+1)} &= -\eta\theta^{(t+1)} \\
&= -\eta \sum_{i=1}^{t} \mathbf{z}^{(i)} \\
&= -\eta(\mathbf{z}^{(t)} + \theta^{(t)}) \\
&= -\eta(\mathbf{z}^{(t)} - \frac{1}{\eta}\mathbf{w}^{(t)}) \\
&= \mathbf{w}^{(t)} - \eta\mathbf{z}^{(t)}
\end{aligned}$$

Algorithm 5 is the Online Projected Sub-Gradient Descent, with mirror function also projecting dual parameter θ to some conditioned set.

Algorithm 5 Online Projected Sub-Gradient Descent

1: **for** $t = 1, \cdots, T$ **do**
2: $\quad \theta^{(t+1)} = \theta^{(t)} - \eta\mathbf{z}$ ▷ $\mathbf{z}^{(t)} = \partial f^{(t)}(\mathbf{w}^{(t)})$
3: $\quad \mathbf{w}^{(t+1)} = \Pi_{\theta \to S} - \eta\theta^{(t+1)}$ ▷ Mirror Projection
4: **end for**

2.1.5 Analysis of Online Gradient Descent

To analyze the regret bound, we first use the general regret bound from OMD

$$R(\mathbf{u}) \leq \psi(\mathbf{u}) - \psi(\mathbf{w}^{(1)}) + \sum_{t=1}^{T} D_{\psi^*}(\theta^{(\mathbf{t+1})} \| \theta^{(\mathbf{t})})$$

As our regularizer is defined as

$$\psi(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|_2^2$$

its convex conjugate ψ^* is also L2 norm. Combining with the definition of Bregman divergence $D_{\psi^*}(\theta^{(\mathbf{t+1})} \| \theta^{(\mathbf{t})}) = \psi^*(\theta^{(t+1)}) - \psi^*(\theta^{(t)}) - \nabla\psi^*(\theta^{(t)})(\theta^{(t+1)} - \theta^{(t)})$. Thus the RHS of OMD regret bound will become

$$R(\mathbf{u}) \leq \frac{1}{2\eta}\|\mathbf{u}\|_2^2 - \frac{1}{2\eta}\|\mathbf{w}^{(1)}\|_2^2 + \sum_{t=1}^{T} \frac{1}{2\eta}\|\theta^{(t+1)}\|_2^2 - \frac{1}{2\eta}\|\theta^{(t)}\|_2^2 - \frac{1}{2\eta}\nabla\|\theta^{(t)}\|_2^2(\theta^{(t+1)} - \theta^{(t)})$$

$$= \frac{1}{2\eta}\|\mathbf{u}\|_2^2 - \frac{1}{2\eta}\|\mathbf{w}^{(1)}\|_2^2 + \sum_{t=1}^{T} \frac{1}{2\eta}\|\theta^{(t+1)}\|_2^2 + \frac{1}{2\eta}\|\theta^{(t)}\|_2^2 - \frac{1}{\eta}\theta^{(t)}\theta^{(t+1)}$$

By completing the square for the summation term

$$R(\mathbf{u}) \leq \frac{1}{2\eta}\|\mathbf{u}\|_2^2 - \frac{1}{2\eta}\|\mathbf{w}^{(1)}\|_2^2 + \sum_{t=1}^{T} \frac{1}{2\eta}\|\theta^{(t+1)} - \theta^{(t)}\|_2^2$$

$$= \frac{1}{2\eta}\|\mathbf{u}\|_2^2 - \frac{1}{2\eta}\|\mathbf{w}^{(1)}\|_2^2 + \sum_{t=1}^{T} \frac{1}{2\eta}\|\theta^{(t)} - \eta\mathbf{z}^{(t)} - \theta^{(t)}\|_2^2$$

$$\leq \frac{1}{2\eta}\|\mathbf{u}\|_2^2 + \sum_{t=1}^{T} \frac{\eta}{2}\|\mathbf{z}^{(t)}\|_2^2$$

By setting up the maximum range of primal space parameter D and maximum gradient size G

$$D = \max_u \|u\|_2 \ u \in S$$
$$G = \max_z \|z\|_2 \ z \in \partial f(\mathbf{w})$$

the above bound becomes

$$R(\mathbf{u}) \leq \frac{D^2}{2\eta} + T\frac{\eta}{2}G^2$$

To minimize the RHS, we can pick optimal η by setting the parital derivative w.r.t. η equals 0.

$$\frac{\partial \frac{D^2}{2\eta} + T\frac{\eta}{2}G^2}{\partial \eta}$$

$$= \frac{-D^2}{2\eta^2} + \frac{TG^2}{2} = 0$$

$$\eta = \frac{D}{G\sqrt{T}}$$

Plug in back to the original equation

$$R(\mathbf{u}) \leq GD\sqrt{T}$$

2.2 Online Normalized Exponentiated Gradient Descent

If we define the regularization function and the loss function in OMD as:

$$\psi(\boldsymbol{w}) = \sum_{k=1}^{K} w_k \log w_k \quad \boldsymbol{w} \in \mathbb{S}^K$$

$$f(\boldsymbol{w}) = \left\langle \boldsymbol{w}, \boldsymbol{\theta} \right\rangle$$

We call this online normalized exponentiated gradient descent (ONEGD). Then the prediction rule becomes:

$$\boldsymbol{w}^{(1+t)} = \arg\min_{\boldsymbol{w} \in \mathbb{S}^K} \left\langle \boldsymbol{w}, -\boldsymbol{\theta}^{(\boldsymbol{t+1})} \right\rangle + \sum_{k=1}^{K} w_k \log w_k$$

Let's add the simplex constraint to objective and call the objective as Lagrangian:

$$L = \left\langle \boldsymbol{w}, -\boldsymbol{\theta}^{(\boldsymbol{t+1})} \right\rangle + \frac{1}{\eta} \sum_{k=1}^{K} w_k \log w_k + \lambda(1 - \sum_{k} w_k)$$

We do first derivative to find the minimizer for linear loss and entropic regularization.

$$\frac{\partial L}{\partial w_k} = -\theta_k + \frac{1}{\eta}(1 + \log w_k) - \lambda$$

$$0 = -\theta_k + \frac{1}{\eta} + \frac{1}{\eta} \log w_k - \lambda$$

$$\rightarrow w_n = \frac{\exp(\eta\theta_k)}{\exp(1 - \eta\lambda)}$$

The dual parameter update is as below.

$$\theta^{(t+1)} = \theta^{(t)} - \eta \boldsymbol{z}^{(t)}, \boldsymbol{z}^{(t)} \in \partial f^{(t)}(\boldsymbol{w}^{(t)})$$

Since mirror function enforces the geometry of the problem, the mirror function becomes

$$g(\boldsymbol{\theta}) = \frac{\exp(\eta\boldsymbol{\theta})}{\sum_{n'} \exp \eta\theta_{n'}}$$

Algorithm of ONEGD is shown in Algorithm 6

Algorithm 6 Online Norm-Exp-GD

for $t = 1, \cdots, T$ **do**
  $\theta^{(t+1)} = \theta^{(t)} - \eta \mathbf{z}$, $\mathbf{z}^{(t)} = \partial f^{(t)}(\mathbf{w}^{(t)})$ ▷ Dual Parameter Update
  $\mathbf{w}^{(t+1)} = \Pi_{\theta \to S} - \eta \theta^{(t+1)}$ ▷ Mirror Projection
end for

It is worth noting that the update of ONEGD is the same update as the weighted majority algorithm. We can have this conclusion from

$$w_n^{(t+1)} = \frac{\exp(\eta \theta_n^{(t+1)})}{\sum_{n'} \exp(\theta_{n'})}$$

$$= \frac{w_n^{(t)} \exp(-\eta z_n^{(t)})}{\sum_k w_k^{(t)} \exp(-\eta z_n^{(t)})}$$

$$\to w_n^{(t+1)} \propto w_n^{(t)} \exp(-\eta z_n^{(t)})$$

Lastly, we show Hedge algorithm in Algorithm 7, which is an unnormalized exponentiated gradient descent algorithm.

Algorithm 7 Hedge Algorithm (β)

$\mathbf{w}^{(1)} \leftarrow \{w_n^{(1)} = 1\}_{n=1}^N$ ▷ Weight initialization
for $t = 1, \cdots, T$ **do**
  RECEIVE $(\boldsymbol{x}(t) \in \{-1, 1\})$
  $i \sim \text{MULTINOMIAL}(\boldsymbol{w}^{(t)} / \Phi^{(t)})$
  $\hat{y} = h_i(\boldsymbol{x}^{(t)})$ ▷ Expeceted 0-1 loss is a linear loss
  RECEIVE $y^t \in \{-1, 1\}$
  $w_n^{(t+1)} = w_n^{(t)} e^{-\beta(\mathbf{1}[y^{(t)} \neq h_n(\boldsymbol{x}^{(t)})])}$ ▷ Exponential update comed from entropic regularization
end for

References

[1] A. Cauchy. Méthode générale pour la résolution des systèmes d'équations simultanées. *C.R. Acad. Sci. Paris*, 25:536–538, 1847.

[2] B. McMahan. Follow-the-regularized-leader and mirror descent: Equivalence theorems and l1 regularization. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 525–533. JMLR Workshop and Conference Proceedings, 2011.

[3] H. Robbins. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 2007.